

# 3D CONSTRUCTION KIT USERS GROUP

NEWSLETTER Oct/Nov 1991  
Issue No. 3



*Hints, Tips and much more...*



## CONTENTS

|                      |         |
|----------------------|---------|
| EDITORIAL.....       | PAGE 4  |
| LETTERS.....         | PAGE 5  |
| HALL OF FAME.....    | PAGE 11 |
| VARIABLES.....       | PAGE 12 |
| 16-BIT ROUTINES..... | PAGE 21 |
| 8-BIT ROUTINES.....  | PAGE 25 |
| PROBLEM PAGE.....    | PAGE 28 |
| HINTS AND TIPS.....  | PAGE 30 |

\*\*\*\*\*

COPYRIGHT on anything which does not carry the writer's name belongs to the User Group. Copyright on all other items lies with the writer. Members whose work is published in the newsletter continue to hold copyright on it.

CORRESPONDENCE. All correspondence addressed to the User Group will be assumed to be for publication unless otherwise stated. If you wish for a personal reply to a letter then please enclose a S.A.E. otherwise replies will be published in the pages of the newsletter.

CONTRIBUTIONS are always welcome from members. Hints, tips, routines or anything else that you think would be of interest to other members are most welcome. It is your club so we want to hear from you. If there is anything in particular that you would like to see covered in the newsletter then please let me know.

\*\*\*\*\*

## HELPLINES

### POSTAL HELP

If you wish to write in for help then please feel free to do so but please don't forget to enclose a stamped, self-addressed envelope and I will do my best to solve the problem.

### TELEPHONE HELPLINE

0492 877305

If you need to telephone me for help then I will try to be available during the hours between 2pm and 7pm each weekday and will be delighted to hear from you and will, of course, do my very best to help with any problems you may have. As this telephone helpline is available only to members of the User Group, please do have your Registration Number handy and give your name when you call.

### FACSIMILE

If anyone finds it more convenient to send a Fax then please feel free to do so. The telephone number is the same as that for the telephone helpline. Requests needing reply should be by letter enclosing SAE.

Please address all correspondence to:

Mandy Rodrigues  
The 3D Construction Kit User Group  
67 Lloyd Street, Llandudno, Gwynedd, LL30 2YP

The 3D Construction Kit User Club is an independently run club but is under license from Incentive Software.



## EDITORIAL

Welcome to Issue 3 of the 3D Construction Kit User Group Newsletter.

Goodness me, it certainly doesn't feel as if half a year has gone by already! Doesn't time fly?! But it does mean that games created by you with the Kit are starting to be produced now. This month I have been lucky enough to get the chance to see some of them. Three of them stand out in particular as being very professionally produced. They are PLANET OF THE DALEKS by Steve Hilder and RESCUE and RESCUE 2 (the sequel) by Paul Ramsell. All of these were produced on the Amiga - I will, no doubt, be seeing games produced on other machines very soon. Planet of the Daleks is not on general release yet but I will let everyone know the details when I receive them. Rescue and Rescue 2 are available from Deja Vu Licenceware, 25 Park Rd, Wigan, Lancs, WN6 7AA at £3.50 (UK), £3.75 (Europe) and £4.(R.O.W). I really recommend that all Amiga owners get Rescue and Rescue 2 if they possibly can. Not just because they are very enjoyable games but they demonstrate just what can be achieved with the Kit. There are excellent loading screens, instructions screens, sound FX or music (very pleasant it is too!). The standard of all three games is excellent!

I had a suggestion this month that I will pass on to you all to see what you think - after all, this is your Club so your wishes are my commands. The suggestion was for the formation of a telephone helpline, where any members wishing to offer their help or advice to other members and exchange ideas etc, be included in the Newsletter. If you think this is a good idea then let me know. If anyone would like to offer their services then all you have to do is to write in with your name, address, telephone number (if you wish to offer telephone rather than postal help) and your computer, and I'll publish them with pleasure. This would also be of immense benefit to overseas readers if they had contact nearer at hand. I think it would also make the User Group extremely friendly if members had direct contact with each other. Anyway, its up to you, let me know what you think.

Quite a few overseas readers have sent me stamped, self addressed envelopes with stamps from their own countries but these are of no use to me at all. If you ask your post office to supply you with an International Reply Paid Coupon (Coupon-Response International or Union Postale Universelle) that will be just right. Speaking of stamps, I have had so many letters from all over the world that I now have three carrier bags full of them and our house is fast becoming a phil... philat... stamp collectors paradise! I am now getting around 35 letters a day - keep on writing please, I love to hear from you - but for obvious reasons there may be a delay in replying to them all so, if you are waiting to receive a reply from me please be patient. I will reply to them all as soon as possible. Reading your letters, especially if they contain routines/contributions for the Newsletters, is the main pleasure for me in running the User Group. You are all so friendly that I wish I could convey this warmth and friendliness within the Newsletters so that you can all share in it.

Well, that's about all from me for this issue. I will be seeing you all again in December with Issue 4 when I hope to be able to send Christmas greetings all around the world. I hope that by that time I will be able to tell you about more games I have received from you!

*Mandy*

## LETTERS

*Dear Mandy*

I find the 3D Kit package quite interesting, but I have a few queries: The rear cover of the newsletter (Issue 1) shows a picture of the menu bar which contains the word "PRINT", but my menu does not contain this word. How do I get round wheels as on the car shown on the box? In the June issue of CU Amiga, there was an advert for the 3D Kit in which a user described how he made a game and used one of the control panels from several included in the Kit. How many borders should be on the disc? Mine only has two (KG and DR). When I use LOAD OBJECT from the menu, I find one object already there (Chopper), which when loaded creates a picture of a helicopter. Is this correct? Finally, when I play the Kitgame, I cannot find a way to open the treasure chest on the Desert Island. I did try removing the lid from the chest, using the editing controls but then I could not enter the chest to descend the staircase as shown in the video. Can you help? I have a 16 bit version of the Kit.

*A. G. Simpson, Walthamstow*

I'm sorry if I have confused you, Mr Simpson, but when I was designing the cover of the first Newsletter I used a screen shot of one of the very early versions of the Kit (before it was finally produced). Don't worry, nobody has missed out, there is no PRINT command on the versions that were finally released. The way that "round" wheels are produced is by using a lot of small inverted pyramid shapes which, when joined together, do give a round effect. There are only two borders provided with the Kit. Yes, the Chopper is an object provided on version 1.0 and 1.1. The way into the chest on the Desert Island is to find the key which is hidden at the top of the palm tree. Just "bump" yourself up the tree to "climb" it.....Mandy

*Dear Mandy*

I really enjoyed reading the first Newsletters. Especially the bits about animation and special programming hints for better effects, those were extremely interesting. I hope there will be a lot more animation hints and algorithms in the next newsletters. It's a pity there were so few pages and hope that, as you suggest, more readers send in routines to pad out the pages of future issues. I am enclosing some hints and tips that I would like to share with everyone that I discovered whilst creating my outer space game called Robolab.

*Jurgen Thurow, Germany - ATARI ST*

Yes, Jurgen, the more hints and tips, routines etc that I receive will determine the size of the Newsletters. You will see that it has already grown from Issue 1. It not only makes life easier for me in producing the Newsletter but a wider range of ideas is so much better. If it was just my ideas all the time the Newsletters would become uninteresting very quickly indeed. All members are invited to send in hints etc, even a very small one is welcome and will be used. When a package is as new as this one all information is valued by the other members.....Mandy

*Dear Mandy*

Just got the first newsletter, liked it, though the stuff on animation wasn't a lot of use. I hope the 8 bitters do get a reasonable say in later issues. You asked for comments on the Kit. I have a few. It's brilliant, easy to use and very versatile. Since I got my copy I haven't done anything else on my computer and I've had it nearly a month. There are three faults with the Kit and the manual is one of them. It is just a list of functions, with insufficient explanations and no tutorial. Saving is where the most serious bug occurs. Very often the kit will lock up when you try to save. When it happens the



only thing you can do is reset, reload and start again from a previous save. The more you save the more it happens. The idea of the PD Library is excellent. I sent my first game off to Robot PD. This was so people who don't have the Kit can see what it can do. Is it OK to send games off to both yourself and ordinary PD libraries? Lastly a question, can you transfer the value of one variable to another?

**John Hilton, Bristol - AMSTRAD CPC6128**

with a strictly limited number of pages to work to it wasn't possible to go into greater detail on so many aspects of the Kit as I would have liked in the manual. The problem with saving on the Amstrad version has been noted and the programmers are working on a solution to this. As soon as I get any further information I will let everyone know. I appreciate just how frustrating this must be but I am confident that it will be sorted out as soon as possible. Actually I have had very little response for the PD library so far and may have to drop the idea. However, there is no reason why people cannot send their games to both the User Group PD and other PD Libraries but do make sure that they are not Licenceware as in that case it would be a breach of contract to do so.....Mandy

#### **Dear Mandy**

I am writing to let you know what I like about the 3D Construction Kit. It's amazing just what this program can get my 128K Spectrum to do, (I thought this sort of thing was only possible on those very expensive dream machines, that you see at computer shows), but this has proved me wrong. I thought including a demonstration video with the software was also a great idea, as getting to grips with software can be difficult at first. It not only shows how it is done, but also gives you ideas for your own environments and was very interesting to watch in its own right. Also I am sure joining the user group will be of great help to me. The first edition (June 1991) Newsletter was very interesting to read, and I found it has given me ideas even if it was all based on 16 bit versions of the Kit. I would like to take this opportunity to wish you the best of luck with the User Group.

**J.P. Messenbird, Mitcham - SPECTRUM 128K**

Thank you for your good wishes, they are appreciated. I am delighted that you are enjoying the Kit so much. The Tutorial Video was a Domark Promotion and was not available with the Computer Club versions of the Kit. I'm afraid, it is not possible to obtain the video separately in case anyone was wondering about it.....Mandy

#### **Dear Mandy**

I just thought I would write a few lines once again, without being a pest, as I know you must be kept quite busy. If you remember when I last wrote, I explained the problem I was having when using my Kit program, it crashed without warning. Well, just to let you know, I wrote to Incentive first, explaining the problem and I received a nice letter back from Anita Walker (Office Administrator) asking me to return my faulty disk. This I did and within a short time I received an upgraded version (01.200) of the 3D Kit, minus the bugs, and even my original postage was refunded. There is a text file on this version which explains the problems and improvements made. Also version V01.200 includes a few more objects and, believe it or not, a type of naughts and crosses game. What more can I say? CONGRATULATIONS and WELL DONE Incentive!

**Gordon Barr, Kilmaronock**

#### **Dear 3D Construction Kit User**

I thought I would just drop you a line via the User Club Newsletter to thank you for your support and to say that I hope you've found the Kit

everything you wanted it to be. If you have any bright ideas on how we could improve it in the future I'd be delighted to hear them. I thought you'd like to know that we're planning to publish a library disk of 3D objects for use in the Kit. If you've designed any 3D objects which you feel are particularly spectacular and would like them to be featured on the disk, please send them to me at Domark's address, or to Mandy, and we'll do our best to include them. The objects can be modelled on any subject - we only ask that they're interesting and well designed. (a full credit and free Domark product will be given to creators of included objects). Lastly, we've just released a compilation of 4 of Incentive's best 3D games called VIRTUAL WORLDS, featuring Driller, Total Eclipse, Castle Master and The Crypt. It's in the shops now so if you missed out on any of the games at the time, here's your chance to catch up. A treat not to be missed, I think. Bye for now, Look forward to hearing from you.

**Matthew Tims, Publisher, Domark**

Domark Ltd, Ferry House, 51-57 Lacy Rd, Putney, London, SW15 1PR

#### **Dear Mandy**

Having just recently acquired my Amiga and 3D Kit, I'm really looking forward to learning as much as I can about my computer and the Kit which I feel I will with yours, and other members help. Also I wish the club every success in the future. I would be grateful if you could help clarify some points for me. I'm a bit unsure about Variables, say variable 40, to do a particular function, will that variable be forever set to that function or can it be re-used differently in a different game, environment etc? If I have a door, which I've instructed to close by activating a rectangle on a pillar, how can I instruct the door to open (or the other way round) by activating the same rectangle? Do I need some sort of a variable?

**David Spencer, Manchester - amiga**

When you create a new environment all the variables are cleared ready for use each time. It is only during a particular game that variables continue to store numbers etc so don't be afraid of using them. Each time you load in the Kit the slate is wiped clean, ready for work on a completely new environment. If you load in a datafile from a previous environment then the variables will be automatically set to the values you set in that environment only. The first two Newsletters have articles on the use of variables together with some examples for you to use. This Newsletter is no exception and contains the final article on variables, this time from an 8 bit users point of view to complete the set. I hope you find them interesting and that they will clear up any fears users may have about them. In answer to your other question, you don't need a variable to open and close the door using just one trigger. What you need is to simply ask the system to check if the door is open or closed and to act accordingly when the trigger is activated (the rectangle pressed). This can be done as follows:

```
IF ACTIVATED? ... if our trigger on the pillar is pressed.
AND VIS? (7) ..... and the open doorway (7) is visible.
THEN INVIS (7) ..... make the open doorway (7) invisible.
VIS (8) ..... make the closed door (8) visible.
ELSE IF SHOT? ..... if our trigger on the pillar is pressed.
AND VIS? (8) ..... and the closed door (8) is visible.
INVIS (8) ..... make the closed door (8) invisible.
VIS (7) ..... and make the open doorway (7) visible.
ENDIF
ENDIF
ENDIF..... there are 3 IF's so we have 3 ENDIF's.
```

I think you will find that this will do the trick nicely. Don't forget to make the open doorway invisible both initially and currently via



the attributes before you start.....Mandy

**Dear Mandy**

I am very pleased with the results of my 3D Construction Kit and would like to know if the RUNTIME MODULE is ROYALTY FREE and if games created with the Kit can be released as Shareware?

**Mr R. Stirling, N Ireland**

The information you are looking for can be found on page 16 of the 16 bit manual (at the end of the "MAKE" explanation) and the second to last paragraph of page 44 of the 8 bit manual (Using the Compiler section). It says "Free distribution of stand alone runnable data files is permitted providing that the 3D Construction kit is acknowledged on screen and in any accompanying documentation". In other words, you can do as you wish with games you create but to give acknowledgement that you used the 3D Construction Kit - something along the lines of "This software was produced with the aid of The 3D Construction Kit".....Mandy

**Dear Mandy**

Many thanks for replying to my letter re my query on loading borders into the Compiler of 3D Kit. I have loaded the Kit game since writing to you and made a stand-alone game with Border Screen loaded okay. I do agree it seems as if it was a loading problem. I did make a couple of copies of the Kitgame Screen and tried loading these into the Compiler - with the game of course - I think it was an error on my part in the saving level not being correct. The code (or badly saved screen) saved by me sounded as if it was loading into the compiler okay, then the rest of the on screen prompts were followed through until the end - a stand-alone game resulted and worked - but with no border screen. You will probably find that if you had a badly saved screen it will appear to be loading okay and it doesn't seem to corrupt the compiler. I feel that at the border loading stage of the compiler program it would be a good idea if you could press BREAK or ANY OTHER KEY if there was an error in loading and start to reload (as many times as was necessary) instead of having to finish off and save the compiled program as it is, and if you find afterwards that your border hadn't loaded you have to go completely through the whole procedure again from the start - that's the only fault I can find with the compiler. As regards the 3D Kit program generally it is absolutely excellent, I say what a superb program for the Spectrum. Many thanks for Newsletter 2 (Aug/Sept issue) - very interesting. The simulated animation routine plus the other bits by Mr George Dixon were really excellent - it helps a terrific amount, especially with the diagrams, I hope Mr Dixon will provide more routines - keep up the good work. Also your explanation of the use of Variables was very easy to understand, I like your style of writing - so straightforward - thanks. If all the future Newsletters are like this they will be worth looking forward to. It seems from the letters in the Newsletter that the old Speccy doesn't seem to have the troubles that the other machines have. Once again, thanks for your reply and keep up the standards.

**Ron Dungey, London - SPECTRUM**

**Dear Mandy**

I think the 3D Construction Kit has a very good price-quality relationship. You get a superb device for unfolding your own creativity and the newsletter is a brilliant idea to explore the possibilities of the system together. I had a problem with the save and load data part and couldn't figure out what was wrong. But through issue 2 of the Newsletter I found out that I could replace my version

V1.1 for version V1.2 which overcomes this problem. I have worked with CAD programs like AUTOCAD and CADVANCE and have designed 3D objects with them through a wire-mesh system. What I like about the 3D Construction Kit is its direct and fast approach. What I miss are more basic geometrical forms like circular forms (necessary for mobile objects like cars etc), and a degree-wise rotation in all planes; furthermore a relative coordinate system in addition to the absolute C.S. which is helpful in construction, especially if you have made a scale drawing. Another suggestion could be that you have negative forms which you can subtract from positive forms. Like this you can construct an attic with two roof forms - one positive and one negative. I think the continuous colour shading which you can achieve in VGA mode is great. I hope the 3D Construction Kit will develop further with time and will also be able to use extended hardware to its full extent. For example the present resolution is 320 x 200 whereas VGA can use 640 x 480 (this would reduce the sawtooth-effect). Also the use of extended memory would enhance the program. What I am dreaming of in the near future is a system with the graphic images of a "Renderman" program (a program with photo-realistic effects where you can imitate different materials and which has superb light source and reflection options) and the uncomplicated directness of the 3D Construction Kit.

**Daniel Bernoulli, Germany - PC**

Actually, Daniel, you don't have to wait any longer for your dreams to come true. The SUPERSCAPE Virtual Reality Toolkit will be launched this Autumn! This truly amazing system, developed by Incentive/Dimension International has everything you could wish for. There are so many features that I cannot list them all here but to give you some idea here are just a few of them: Superscape features a Fully Interactive Editor which allows views in perspective 3D plan, or side and front elevations with all aspects of objects able to change in real time. Multiple Viewpoints which may be either user controllable, attached to static or moving objects, and may themselves move and track other target objects. Object Control can be either first person viewpoint (eg driving a car) or second or third (eg flying a model aircraft). Multiple Moving Objects with realistic physical movement model incorporating friction, gravity, collisions and restitution. Users may take direct control of objects, they may move along pre-set routes or they may use the in-built programming language for intelligent control. Objects may launch projectiles which can then behave independently. Objects can rotate freely in all three axes, under user control. Object Bending allowing sophisticated movement modelling of objects which are traditionally difficult to animate. Transperancy (including colour tints) is possible and especially suitable for windows, allowing a visual link between interior and exterior spaces. Multiple mobile point light sources together with ambient lighting. All the shapes you could wish for are possible such as circles, cones etc. VR resolution/size equates to size of British Isles with 1mm accuracy. Update speed variable up to 70 frames per second. Image resolution of 640 x 480 x 256 colours. With network capabilities with up to 8 workstations sharing the same virtual reality. Superscape offers much, much more than this and is designed to cater for design in the building industry, landscaping, interior design, research, space planning and creative services such as Cartoon/animation specialists, Film/TV production companies and Advertising agencies. There will be full backup/training and support but, the one drawback for most of us ordinary home computer users, will of course be the price. Superscape systems start at £12,500 but include the hardware ie. Computer (Gulp!). Full details are available from Incentive.....Mandy



**Dear Mandy**

With reference to the last Newsletter, I noticed that there was only one reader with an Amstrad CPC6128 - all the others were for AMIGA, ATARI, mostly!!?; there was one CPC464, where have all the other CPC's got to?, never mind perhaps our numbers will increase as time goes on. I was very interested in the article by David Sambrook and the one regarding a routine for creating more realistic doors and doorways, it has given me an idea, for I am in the process of making a "walk-through" of my flat (and there are many doors to contend with, also cupboard doors). Bearing in mind that my machine is the Amstrad CPC6128, I have a couple of queries regarding the 3D Kit. How does one CAT the disc to see how much space is left for further saving of Progs? As it is I have NO WAY of seeing what I have on disc or how much room I have to spare. How can I get a line vertical? I've tried the TURNING icons in all sorts of ways but still can't do it. (I did manage to do it once but believe it or not I've forgotten how I did it!) Is it possible to make a back-up copy of the 3D Kit? I assure you it will only be for my own use. This is in case of accidental damage to the original disc. I'd be grateful for your advice.

**George Baxter**, Sevenoaks - AMSTRAD CPC6128

A fair proportion of the User Group members own Amstrad CPC's both 464's and 6128's, George, it is just a matter of time before they begin to see more of them through the pages of the Newsletters. The 8 bit versions are a lot "newer" than the 16 bits (these were released a couple of months before the 8 bits), so they haven't had as long to become familiar with the Kit. I'm afraid there is no way to CATalogue the disc under program control but perhaps this feature can be incorporated in future versions - I'll pass the suggestion on to the programmers. Use the POINT in conjunction with the TURN icons to get your line in the position you prefer. You can make one backup copy for your own use - in fact it is recommended in the manual that you do so for safety reasons, just in case you damage your original.....Mandy

**Dear Mandy**

My name is Mieke (Annemieke) Van Der Poll-Vervest. I am a 45 year old woman. When I sent you my registration card I forgot to mention if I was male or female, so I received your Newsletter addressed to Mr Van Der Poll instead of Mrs. At the end of May this year I was staying for a short holiday with my brother and sister-in-law in Wales. Whilst there I bought some computer magazines and in one of them I read about the 3D Kit and immediately wanted to buy it but it wasn't generally available at that time. So it was only when back in Holland in July that I was finally able to buy it. I've been using the Kit ever since and I enjoy every minute working with it. I followed the instructions and started building a house with several rooms in it. I am getting quite familiar with the basics of designing things, going through doors etc and I've started my first little animation, I built a garden with a swimming pool and created a simple figure who is swimming in it. I do not understand the whole language of the Kit yet, especially working with variables, but I think I have about the same problem as you, seeing things too difficult in the beginning and then, after trying out some things for a while I begin to understand that it is all much easier than I had thought. I hope to read a lot of tips in your Newsletters and when I have something of use for others I will send it in. I wish you and your family a lot of success and fun with the program and the User Club. I think making a public domain club is a very good idea and if I am able to make a nice program I would love to share it with others and vice versa.

**Mieke Van Der Poll**, Holland - PC

My apologies, Mieke, for presuming that you were a man. It is

difficult sometimes to get it right especially with names that I am unfamiliar with. I remember waiting 3 years before writing to a gentleman reader of Adventure Probe magazine to ask if he were male or female (I didn't want to insult him) but as he is called Paraskevas Tsourinakis I just had to ask in the end. If I have made a similar mistake with anyone else then please do let me know. We have members all over the world and it isn't always possible to get it right in every case and, as I am not really into women's lib, I follow the old rule of "when in doubt, choose Mr". I am so pleased to hear that you are enjoying using the kit so much. Actually I am sure other members would be interested in knowing how you programmed the animation of the swimmer in your pool so you don't have to wait until you have a suitable routine to send in, as that one will be most informative. In case anyone is wondering about my reference to Adventure Probe, it is a magazine for computer adventure players that I edit (plug, plug). so if anyone would like further details then let me know and I will be delighted to oblige.....Mandy

**Dear Mandy**

I really had to write to thank you for sending me the latest Newsletter - my first sample copy. I hesitated about sending for it at first as I honestly thought that "Newsletter" meant just that - a short letter sized paper with some news on it! How wrong I was, and how pleased I am that I decided to send for a sample copy! I am now going to become a full member of the User Group. The hints and tips, letters, articles - in fact, everything, was most interesting and very useful indeed. Thanks to everyone who sent in the articles etc, I am now beginning to use the Kit to its full potential (well, at least I hope so!). I was particularly pleased with the explanation of ENTRANCES as I too had thought that when you created an entrance it automatically appeared where you wanted it to be and you could just walk through it to another area! I have just one little grumble, I wish that the manual had stated that only 60 objects per area were possible! I couldn't understand what was going wrong with my environment until I sent you my datafile to check out.

**Simon Davidson**, Huddersfield - AMIGA

Yes, I'm afraid that as soon as you create more than 60 objects in any area things can start to go wrong in very peculiar ways, from areas suddenly having their names changed to objects appearing in strange places, in fact, all sorts of things. If anyone experiences strange problems with their datafiles it is always worth checking the number of objects first. There are ways to overcome this and some hints are included in this newsletter on that particular aspect. I am always happy to check out your datafiles provided they aren't send in as stand-alone environments and if I cannot come up with the answer, I know a man who can!.....Mandy

\*\*\*\*\*

## HALL OF FAME

Sincere thanks to the following members who kindly send in contributions for the User Group Newsletter during the past two months:

FERNANDO CARVALHO, JURGEN THUROW, MIEKE VAN DER POLL, GEORGE DIXON, BRIAN WOODLEY, JAMES NEILL, STEPHEN FLANAGAN, PAUL RAMSELL, LEON BOROUKH, J.C.LATTY, A.P.SMITH, GEORGE BAXTER, RICHARD HORN and D.G.MARLEY.

Everyone is cordially invited to send in contributions, no matter how small they might be. Everything helps! Next issue will be in December!



### VARIABLES---VARIATION ON A THEME

Further to Mandy's article on Variables in the last Newsletter (Issue 2 AUG/SEPT) I would like to enlarge upon this and attempt to explain some of the variable manipulation commands. This article is written from the 8-BIT slant, but, might be of some use/interest to 16-BIT users.

#### BACK TO THE BEGINNING

A variable can be thought of as a slot or mailbox within the memory of the computer. This mailbox has for the purposes of the "3D-KIT" a fixed address or variable number between 0 to 127 (8-BIT), but we can only use variables 0 to 111 to store information ourselves. The remaining 16 variables (112 to 127) are known as SYSTEM VARIABLES and are used by the "3D-KIT" control program (SYSTEM) to keep track of such things as the player's viewpoint position within the environment. Although we cannot use these 16 System variables for normal data storage we can still alter the values held in the variables to good effect. Some particularly interesting effects can be created ! All will be revealed later!!!!

#### BIT BY BIT

In the "3D-KIT" (8-BIT), a variable can be regarded as an 8-BIT store capable of holding a number in the range 0 to 255. What is a BIT? Well, a computer can only understand one thing: the presence or absence of electric current which can represent 'YES' or 'NO', 'ON' or 'OFF', or '1' or '0'. This state of being 'ON' or 'OFF', or '1' or '0' can provide a means of counting or representing numbers/values using only 8 digits. When counting in DECIMAL we use the digits 0 to 9. To carry on counting after 9 we must go back to 0 and carry a 1 into the "TENS" column and so on thereafter. We know that  $152 = 1 \times 100 + 5 \times 10 + 2 \times 1$  and we can also write 152 in its expanded form by expressing it as,  $1 \times 10^2 + 5 \times 10^1 + 2 \times 10^0$ . Any number raised to the power of zero ( $10^0$ ) is equal to 1. The BINARY system uses only digits 0 and 1---(great that's all a computer knows-remember?). When we count in BINARY the same rules apply as in the DECIMAL system, but this time, after counting to 1 we go back to zero and carry 1 into the next column left. In binary, each position or column represents a power of 2 rather than a power of 10. The BINARY number 0111 can be written as:-

$$(2 \times 2) + (2 \times 1) + 1 \text{ or } 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 7$$

Each BINARY column position or BINARY digit is known as a BIT (Binary digit). 8 BITS are known as a BYTE.

A variable's 8 binary digits (BITS) are numbered 0 to 7 from right to left, each BIT representing a power of 2. Refer to the diagram below; you should see that as you move from right to left, so each BIT position doubles the previous value.

| BIT NUMBER       | 7     | 6     | 5     | 4     | 3     | 2     | 1     | 0     |
|------------------|-------|-------|-------|-------|-------|-------|-------|-------|
| POWER OF 2       | $2^7$ | $2^6$ | $2^5$ | $2^4$ | $2^3$ | $2^2$ | $2^1$ | $2^0$ |
| POSITIONAL VALUE | 128   | 64    | 32    | 16    | 8     | 4     | 2     | 1     |

If all of the POSITIONAL VALUES are added together, we find that one 8-BIT variable can hold a maximum value of 255 DECIMAL or 11111111 BINARY.

### HIGH AND LOW

This idea can be extended over two variables (TWO BYTES) to allow 16 BITS to represent numbers between 0 to 65535.

i.e.

#### VARIABLE 2

| BIT No.          | 15       | 14       | 13       | 12       | 11       | 10       | 9     | 8     |
|------------------|----------|----------|----------|----------|----------|----------|-------|-------|
| POWER OF 2       | $2^{15}$ | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | $2^9$ | $2^8$ |
| POSITIONAL VALUE | 32768    | 16384    | 8192     | 4096     | 2048     | 1024     | 512   | 256   |

#### VARIABLE 1

|  | 7     | 6     | 5     | 4     | 3     | 2     | 1     | 0     |
|--|-------|-------|-------|-------|-------|-------|-------|-------|
|  | $2^7$ | $2^6$ | $2^5$ | $2^4$ | $2^3$ | $2^2$ | $2^1$ | $2^0$ |
|  | 128   | 64    | 32    | 16    | 8     | 4     | 2     | 1     |

Once again adding all the POSITIONAL VALUES together gives us a maximum value of 65535.

If two variables are used in this way for storage of numbers over 255 then the variable holding values 256 upwards (variable 2 above) is termed the HIGH BYTE and the other variable is termed the LOW BYTE. Because the LOW BYTE of our 16-BIT store can only hold a value upto 255 we need to have the means of working within the HIGH BYTE when we go above 255. If we wish to set the store to a particular value above 255, then we must first determine by how much our parameter value would overrun the LOW BYTE i.e. how much should be stored in the HIGH BYTE. This is easily achieved by division of our value by 256, this being the first POSITIONAL VALUE of our HIGH BYTE or the value at which a binary CARRY is set when counting 1 up from 255. The value obtained from this division is stored in the HIGH BYTE with the remainder being stored in the LOW BYTE.

e.g. We want to store 7000 in variables 1 & 2

$$7000 \div 256 = 27 \text{ (6912)}$$

$$7000 - 6912 = 88$$

Therefore we would store the value 7000 in the following way:-

SETV 88 1 (SET VARIABLE 1 TO 88.....LOW BYTE)

SETV 27 2 (SET VARIABLE 2 TO 27.....HIGH BYTE)

### 8-BIT VARIABLE COMMANDS

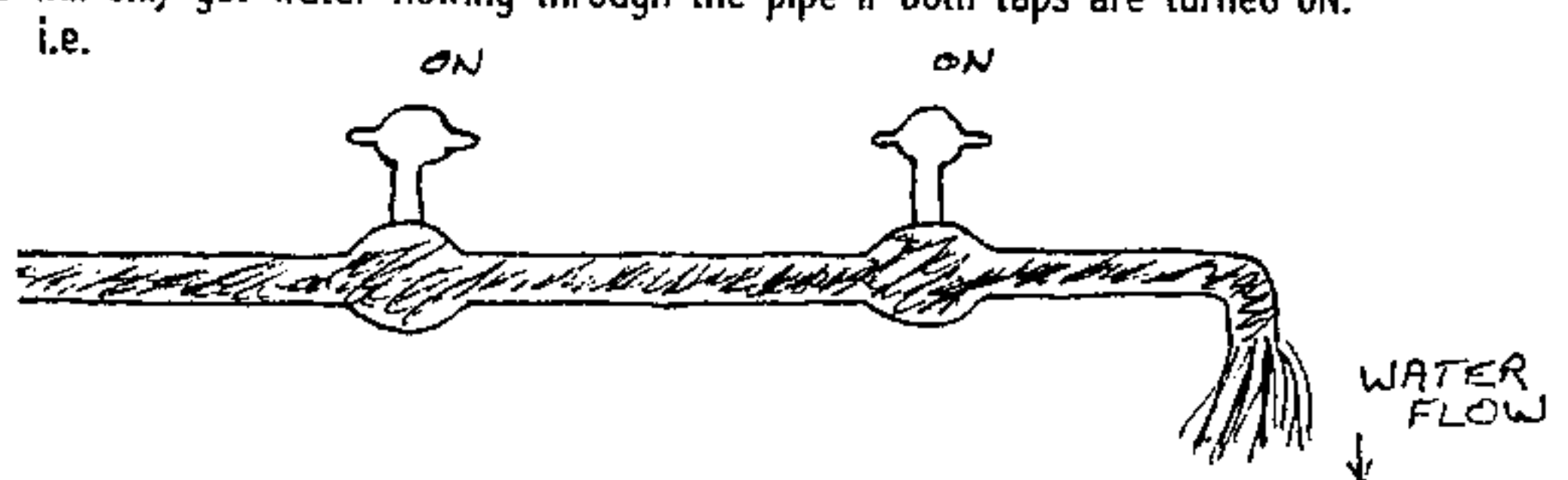
Hopefully, I have not lost too many of you but that you all have got the 'BIT' between your teeth!!!! I feel that an understanding of the construction of a variable is necessary in order to make full use of the '3D-KIT FCL'. I will now try to explain the 8-BIT commands that allow manipulation of the variable values.

8-BIT commands are:- ADDV; ADCV; ANDV; TESTV; CMPV; ORV; SETV; SBCV; SUBV; XORV.

Of these ten commands which can directly access/read or alter the contents of a variable, I will explain the use of four:- ANDV; TESTV; ORV, and XORV. I feel that the remaining commands are explained sufficiently within the '3D-KIT' MANUAL. These four commands rely upon an understanding of the POSITIONAL VALUES of each BIT of a variable as they require these values as their parameter to carry out BINARY LOGIC functions.

#### ANDV

Consider a water pipe with two taps along its length controlling the water flow. You will only get water flowing through the pipe if both taps are turned ON.



This is the basis of the AND function in that you will only get a result of YES(1) if both values are set to YES(1).

The ANDV command is similar to trying to pass current through the BITS of a variable compared to the BITS of a parameter (represented in BINARY). The command looks at each BIT of the variable comparing it to its equivalent BIT (with same POSITIONAL VALUE) in the parameter. If two equivalent BITS (variable to parameter) are set to 1 then a 1 results; if one or both of the two equivalent BITS is set to 0 then a 0 results. The result is stored back into that particular variable BIT position.

i.e. ANDV 8 1

This means that the number 8 is to be AND'ed with variable 1. For this example, let us assume that variable 1 is set to 8 already.

Decimal 8 is equivalent to 1000 Binary.

| BIT NUMBERS          | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------------------|---|---|---|---|---|---|---|---|
| BINARY VALUE OF 8    | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| CONTENTS OF VARIABLE | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| RESULT OF ANDV       | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

If variable 1 were set to some value other than 8, say for instance 64, then the result would be quite different:-

| BIT NUMBERS        | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------------------|---|---|---|---|---|---|---|---|
| BINARY VALUE OF 8  | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| VARIABLE SET TO 64 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT OF ANDV     | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

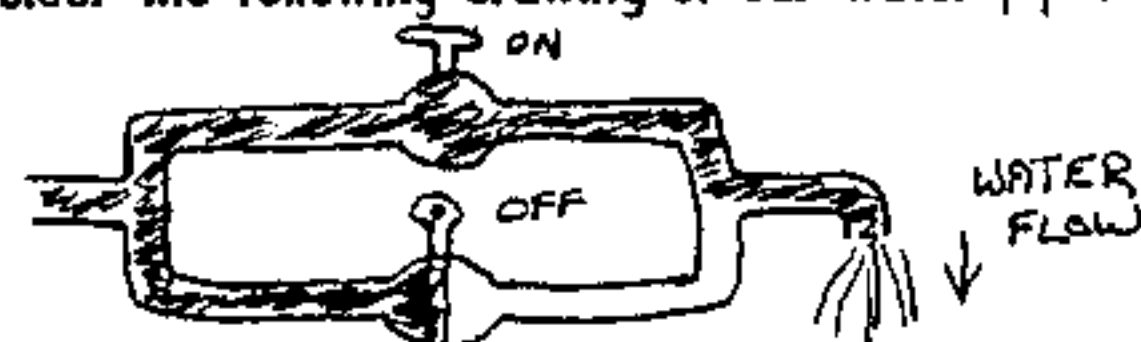
As can be seen from the diagram, using the ANDV command will only return a 1 if two equivalent BITS are set to 1. Because the result is stored back into the variable tested, the previous value of that variable could be corrupted. For this reason I recommend the use of the command TESTV.

### TESTV

This command carries out an ANDV between the parameter and the variable, but in this the result is not stored into a variable but remains as a 1 or 0 to be read by a command such as IFEQ. Such a command is to immediately follow the TESTV command.

### ORV

Consider the following drawing of our water pipe:-



I hope that you can see that water will flow if one OR both of the taps are turned ON. This explains the OR function.

The ORV command simply looks at each individual BIT of the particular variable comparing it to its equivalent BIT of the parameter value you set. A 1 is set in that BIT if either OR both of the two BITS are set to 1.

i.e. ORV 16 1

This means perform a logical OR between the value 16 and variable 1:-

| BIT NUMBERS          | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------------------|---|---|---|---|---|---|---|---|
| BINARY VALUE OF 16   | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| CONTENTS OF VARIABLE | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT OF ORV        | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |

As you can see, this provides a means of setting a 1 or YES into a particular BIT of a variable. This could be used for storing information relating to objects carried say for instance is a certain numbered key carried(?). One variable could store information on upto eight keys. You would simply say that each of the eight BITS of the variable related to keys 1 to 8. As you pick up each key found around the environment, so you would set the relevant BIT in the variable to 1 using the ORV command

i.e.

| BIT NUMBER | 7   | 6  | 5  | 4  | 3 | 2 | 1 | 0 |
|------------|-----|----|----|----|---|---|---|---|
| KEY NUMBER | 8   | 7  | 6  | 5  | 4 | 3 | 2 | 1 |
| ORV VALUE  | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |

It follows that if you were programming for collection of key 6, then you would enter the command:

ORV 32 1 (if using variable 1)

The TESTV command provides the means of reading this information. For example, if we want to check if key 6 has been collected just enter the command:

TESTV 32 1

TESTV 128 1 if you were checking for key 8.

In the last NEWSLETTER (ISSUE 2) I explained a routine for collecting a key and opening a door with the correct key.

### XORV

XORV is a slightly more complicated command in that I cannot think of a simple drawing analogy. All I can say is that you study my diagram below:-

| PARAMETER BIT | VARIABLE BIT | RESULT OF XORV |
|---------------|--------------|----------------|
| 0             | 0            | 0              |
| 1             | 0            | 1              |
| 0             | 1            | 1              |
| 1             | 1            | 0              |

I hope you can see that this command (EXCLUSIVE-OR FUNCTION) only sets a 1 as a result if only one of the two equivalent BITS being compared is set to 1. If both BITS are set to 0 or both BITS are 1 then a 0 is the result.

We can use this command to good effect for toggling FLAGS i.e. switching switches ON and OFF.

e.g.

### SWITCH 8 ON

| SWITCH No.  | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
|-------------|---|---|---|---|---|---|---|---|
| XORY 128    | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| VARIABLE    | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT XORY | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

### SWITCH 8 OFF

| SWITCH No.  | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
|-------------|---|---|---|---|---|---|---|---|
| XORY 128    | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| VARIABLE    | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT XORY | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |



### DEMONSTRATION OF COMMANDS

There now follow some routines that do nothing more than (hopefully) demonstrate the variable commands. When you have programmed in each section, I want you to go to the TEST screen where you will be able to see the results of your variable manipulation in the SCORE window.

#### ANDV/ORV

The aim of this routine is for you to first press key "T", you should see the SCORE set to 8. Now press key "C", the SCORE will now increase to 10.

1. CREATE and EDIT the following LOCAL conditions:

```
CMPV 84 121      (THIS IS LOOKING FOR KEY T TO BE PRESSED 84 BEING ASCII FOR T)
```

```
IFEQ
```

```
THEN
```

```
ORV 8 1          (SET BIT 3 -POSITIONAL VALUE 8- IN VARIABLE 1 TO ON)
```

```
END
```

```
CMPV 67 121      (THIS IS LOOKING FOR KEY C TO BE PRESSED 67 IS ASCII C)
```

```
IFEQ
```

```
THEN
```

```
ORV 2 1          (SET BIT 2 -POSITIONAL VALUE 2- IN VARIABLE 1 TO ON)
```

```
END
```

```
CMPV 68 121      (68 IS ASCII FOR D)
```

```
IFEQ
```

```
THEN
```

```
ANDV 2 1
```

```
END
```

The last condition is looking for key D to be pressed; when it is then the ANDV command carries out a LOGICAL AND of the positional value 2 (BIT 1) with variable 1. As BIT 1 is already set to 1 then a 1 results, but a 0 results in all other BITS. Variable 1 is therefore left with one bit set, this having a positional value of 2 therefore, on the TEST screen you should see the SCORE go to 2.

#### XORV

The purpose of this routine is to demonstrate XORV. You are to press key "E". Upon the first press you should see the SCORE set to 64; each subsequent press of the "E" key will TOGGLE the SCORE between 0 and 64.

```
CMPV 69 121
```

```
IFEQ
```

```
THEN
```

```
XORV 64 1
```

```
END
```

#### NOTE:

Users may find that adding the command DELAY 10 after the actual XORV command will improve the routines. If a delay isn't used you may find that the XORV will work so fast as to make the score too difficult to read. It would be as well to remember this when using any XORV command in your routines.

#### TESTV/ADDV

This routine requires that you press key "T". Upon each key press then 1 is added to variable 1, you should see the SCORE increase by 1. The TESTV commands are looking at variable 1 testing for particular BITS to be set. Messages on the screen will inform you of the BITS set!

```
CMPV 84 121
```

```
IFEQ
```

```
THEN
```

```
ADDV 1 1
```

```
ENDIF
```

```
TESTV 1 1
```

```
IFEQ
```

```
THEN
```

```
ELSE
```

```
TEXTCOL 3
```

```
PRINT 2 10 4
```

```
ENDIF
```

```
TESTV 2 1
```

```
IFEQ
```

```
THEN
```

```
ELSE
```

```
TEXTCOL 3
```

```
PRINT 3 10 5
```

```
ENDIF
```

```
TESTV 4 1
```

```
IFEQ
```

```
THEN
```

```
ELSE
```

```
TEXTCOL 3
```

```
PRINT 4 10 6
```

```
ENDIF
```

```
TESTV 8 1
```

```
IFEQ
```

```
THEN
```

```
ELSE
```

```
TEXTCOL 3
```

```
PRINT 5 10 7
```

```
END
```

CREATE and EDIT the following messages:

```
MESSAGE 2:BIT NOUGHT SET
```

```
MESSAGE 3:BIT ONE SET
```

```
MESSAGE 4:BIT TWO SET
```

```
MESSAGE 5:BIT THREE SET
```

Now for the interesting BIT where we combine all that we have "learnt(?) so far into what I think is an exciting routine. It's not just the 16-BIT users who can have automatic player movement, I have managed to crack the secret of the "SYSTEM VARIABLES" and bring to you at no expense:

#### AUTOMATIC MOVEMENT OF PLAYER ACROSS AN AREA

##### Z-AXIS/X-AXIS

The X and Z axes have viewpoint ranges of from 0 to 8191. Object positions available are 0 to 127 (128 positions). One unit in object position is equivalent to 64 area viewpoint positions.

##### Y-AXIS

The Y axis has viewpoint range of from 0 to 4093. This provides object positions of 0 to 63 (64 positions).

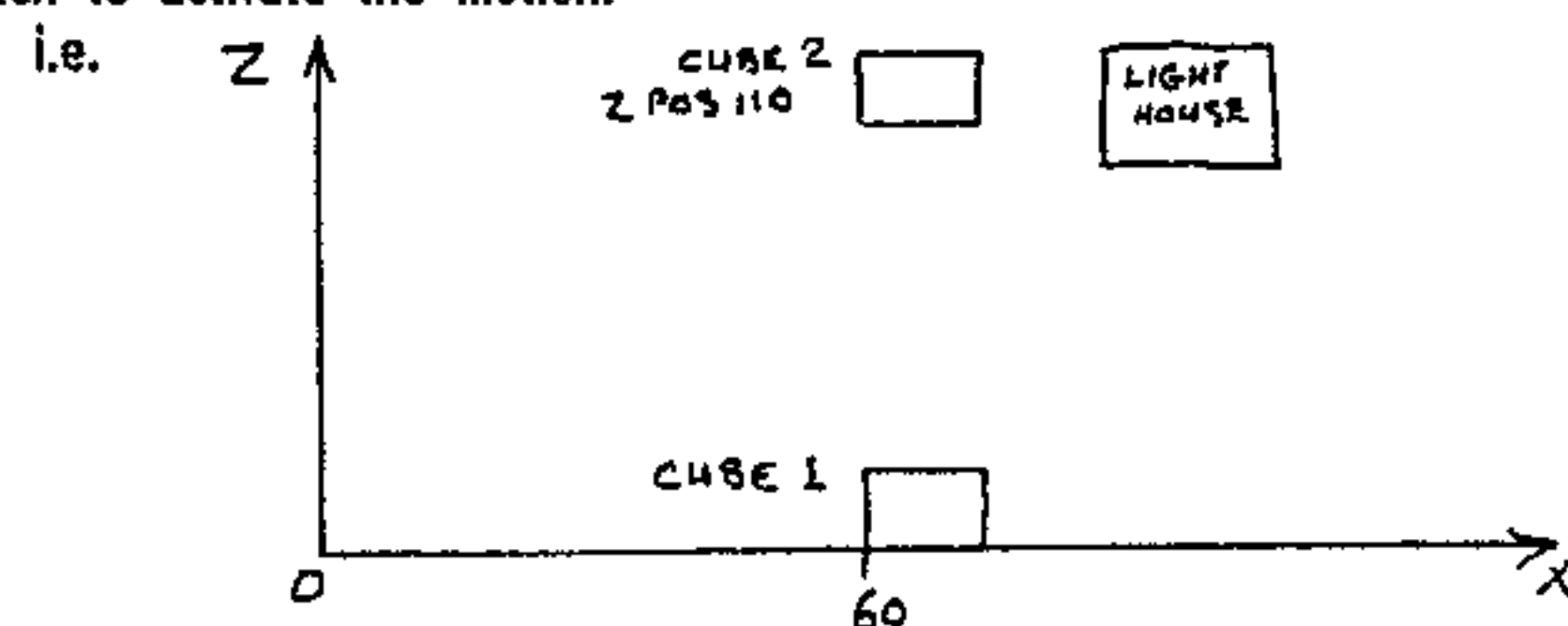
##### VIEWPOINT VARIABLES

The SYSTEM VARIABLES 112 to 117 keep track of the Player's viewpoint throughout the environment and can be easily programmed to alter the position of the player.

##### ROUTINE FOR Z-AXIS MOVEMENT

The object of this routine is to demonstrate how to achieve movement across the area in the Z-AXIS. Additionally it demonstrates several other features achieved by use of the SYSTEM VARIABLES.

The routine requires that the player step onto a flat block (CUBE 1 below) and be automatically transported across the area to CUBE 2 position. The player has to press a switch to activate the motion.



1. CREATE cube (object 2) and shrink in height to size 1. Position this cube at X=60, Z=000. This is cube 1 on the sketch.
2. COPY this cube and position at X=60, Z=110.
3. CREATE a rectangle and flip it to enable positioning flat on top of cube 1. This will provide a trigger pad. Make this rectangle INVIS via the SHADES edit function (object 4).
4. CREATE a cube and position alongside CUBE 2. Increase height of this cube (object 5) to say 16. This will be our "LIGHTHOUSE" and will give a sense of perspective to the movement.
5. Set cube 2 (object 3) to CURRENT and INITIAL states of INVISIBLE via ATTRIBUTES.
6. CREATE and EDIT the following LOCAL conditions:

1. IFHT 4  
THEN  
TEXTCOL 3  
PRINT 2 4 7  
END
2. CMPV 71 121 (VARIABLE 121 RECORDS KEY PRESSES--71 IS ASCII CODE FOR "G")  
IFEQ  
THEN  
ORV 1 1 (WE ARE SETTING A FLAG IN VARIABLE 1 TO INDICATE KEY G PRESSED)  
END

3. CMPV 28 117 (THIS IS Z POSITION HIGH BYTE. WE WANT THE ACTION TO STOP WHEN WE REACH OBJECT POSITION 112. THIS EQUATES TO VIEWPOINT POSITION 7168. DIVIDE THIS NUMBER BY 256 TO CALCULATE HOW MUCH IS SEEN IN THE HIGH BYTE i.e. 7168÷256=28)  
IFEQ  
THEN  
SETV 0 1  
INVIS 2 (MAKE CUBE INVIS SO IT LOOKS AS IF WE HAVE MOVED ON IT!!)  
VIS 3 (MAKE CUBE 2 VIS)  
SETV 7 118 (THIS GIVES 35° VIEW ROTATION DOWN)  
ELSE  
ORV 2 1  
ENDIF

4. CMPV 3 1  
IFEQ  
THEN  
SETV 0 118 (THESE NEXT THREE LINES SET THE SYSTEM VIEWPOINT ROTATION VARIABLES TO 0 TO MAKE THE PLAYER LOOK STRAIGHT AHEAD.)  
SETV 0 119  
SETV 0 120  
SETV 16 113 (BRING PLAYER TO X-OBJECT POSITION 64 i.e. CENTRE OF BLOCK)  
SETV 0 112 (DITTO)  
ADDV 64 116 (MOVE PLAYER BY 1 OBJECT POSITION STEP AT A TIME)  
ADCV 0 117 (ADDING CARRY IF THERE IS ONE TO Z-POSITION HIGH BYTE)  
SYNCSND 2  
REDRAW (THIS IS ESSENTIAL AFTER EACH MOVE TO ENABLE UPDATE OF CHANGES)  
END

CREATE MESSAGE 2: "PRESS KEY G TO START THE MOTOR"

Faster movement can be obtained by editing condition 4 such that ADDV is either 128 (2 object positions), 192 (3 object positions) to variable 116 (Z-position LOW BYTE). Faster movement still can be obtained by deleting line ADCV 0 117 and editing line ADDV 64 116 to read ADDV 1 117 (add 1 to Z-position HIGH BYTE). This will give a speed increase of four times the original.

*George Dixon*

AMSTRAD CPC6128

My thanks to George for a most interesting and in-depth look at how variables work and for the excellent and useful routines. 16 Bit users will also find them useful as only a little adaptation is needed to convert the commands/routines to the 16 bit machines.



## HOW TO GET YOUR GAMES NOTICED

By Mandy

You've completed your masterpiece. You are as proud as can be. You have a game ready to launch onto the unsuspecting public that is going to knock them for six! Problem is...err... what to do next? How to go about it?... Read on.

Most people find to their amazement that the actual programming of their games is the easy part. It is when they enter the world of marketing, glossy magazine reviews and so forth that they realise just what a complicated business it can be. It is especially upsetting when you submit your game to a software company and they return it almost immediately (that is if they bother to return it!), with a polite (or sometimes not so polite), note saying "Thanks, but no thanks!" Especially when you know that they cannot possibly have had time to explore your game properly. Here are a few hints and tips that might just help to get your game that "second look".

When submitting a game to a software house, or a reviewer in a glossy magazine, you should bear in mind the following. The person who is going to take a look at it is not going to play it in the same way as you or I would. You need to provide something to catch their interest. It is a good idea to have the first few areas "open" to reviewers and testers. There is nothing more off-putting, especially if you are a reviewer, than to enter the first area and immediately become stuck with a difficult problem to solve. They quickly become bored and the game is put to one side. So make those first areas as interesting as possible with lots to see and lots to do. You can leave your lovely mindbending problems until later on in the game. In this way the reviewer will have plenty of room to explore and to get an idea of how well your program works. Then his interest might be sparked and he will spend hours working on it.

It is also a good idea to have one of your initial areas very well drawn and designed with nice colourful objects etc., because the magazine may well decide to publish a screenshot and it is usually one of the very first areas that are selected.

Packaging is also important, and I don't mean beautiful colourful plastic cases with maps, 100 page booklets and so forth. They are, of course very nice but I am talking about presentation. Make sure that you provide a solution, or at least some hints and tips to help the reviewer along the way. A storyline of what the game is all about with the player's objective and as much information as you can think of that might be of help.

But, before you even consider sending your game to a reviewer or software house, you must make sure that it has been playtested properly by more than one person. Running through the game yourself is no good at all. You know what you have to do and bugs only show up when someone tries to do something that you haven't planned for them to do! So someone who has never seen your game previously is the ideal person to spot the errors. Many people will be more than happy to playtest your games. Ask your friends, certainly, but try to find someone who has had experience in evaluating games if at all possible. These are just some ideas to bear in mind. If you would like to know more then let me know and I will also be happy to put you in touch with someone who can playtest for you if you wish.

\*\*\*\*\*

## 16 BIT ROUTINES

### FOLLOW MY LEADER!

By Brian Woodley on ATARI ST

I have written a little routine which causes an object to chase the player's view point around and if it catches them then various effects can happen. The first thing to do is to create an object to chase you around, so CREATE a cube and mark it as MOVEABLE in its ATTRIBUTE menu. This is object (2). You should also make a note of its start coordinates. Two variables are used V40 and V41, these hold the X and Z coordinates of the block. Enter this in the first GENERAL CONDITION

SETVAR (X,V40)

SETVAR (Z,V41)

Substitute the starting X and Z coordinates of the block for the X and Z in the commands above. Now enter the following commands in the AREA conditions:

IF VAR<? (V40,V0)

THEN ADDVAR (20,V40)

ENDIF

IF VAR>? (V40,V0)

THEN ADDVAR (-20,40)

ENDIF

IF VAR<? (V41,V2)

THEN ADDVAR (20,V41)

ENDIF

IF VAR>? (V41,V2)

THEN ADDVAR (-20,V41)

ENDIF

STARTANIM (1)

Create animation 1 and enter:

INCLUDE (2)

MOVETO (V40,20,V41)

RESTART

This is assuming that the starting Y coordinate of the block is 20. If it isn't then change the 20 in the animation to whatever it is. Remember to set the initial condition in the DEFAULTS to 001. Press RESET and the cube should be chasing you around like a maniac dalek!. If it isn't then check your typing because it works for me! To speed it up or slow it down, change the 20's in the area condition, increase it to speed it up or decrease it to slow it down (minimum 1), but be warned, the faster it gets, the jerkier it gets. This is great but it is a bit boring, lets make it buzz when it catches you! Enter this as a condition on the object:

IF COLLIDED?

THEN SYNCSD (6)

ENDIF

Now turn up the volume and let it catch you. All that is left now is to add arms, legs etc and change the INCLUDE number in the animation to use this group. Change the condition so that instead of just making a noise it drains your energy or similar and voila, you have a moving guard to foil players. There is one drawback to this routine, the guards will walk through your walls. Perhaps they are not guards but ghosts! (Yeah, that should work!). I'll leave it up to other geniuses to work that problem out!

### LOADING SCREENS AND MUSIC!

By Paul Ramsey 11 - AMIGA

Here are some tips on how to include a music soundtrack and IFF loading screens into a finished Kit game. Members will need the following programs/files in order to create the startup-sequences that

I will describe below, but all are available on Public Domain disks and are therefore very cheap:

MED V3.0 - Music program  
 MEDPLAYER - MED V3.0 file player, should be on MED program disk  
 SHOW - IFF picture displayer  
 UNSHOW - turns off IFF picture  
 CLS - in C directory of Workbench disk  
 RUN - in C directory of Workbench disk

First you must create two IFF pictures with Dpaint III or a similar graphics package. The first picture will be your startup screen so do a drawing that will portray the mood of your game. Save this picture to a blank formatted (and installed) disk as PIC1. The second picture ought to give a brief scenario and instructions (in suitable text font) for your game, save this to your disk as PIC2. Now using MED V3.0, compose (or if you are as musical as I am, get someone else to compose) a suitable soundtrack and save the tune onto your disk as SND1. Then copy the MedPLAYER file that comes with MED V3.0 onto the disk. Next create a C and S directory on the disk and copy the files RUN, CLS, SHOW and UNSHOW into the C directory and then save the following into the S directory as STARTUP-SEQUENCE using a text editor:

```
CLS
SHOW PIC1
SHOW PIC2
UNSHOW PIC1
RUN MedPLAYER SND1
GAMENAME
```

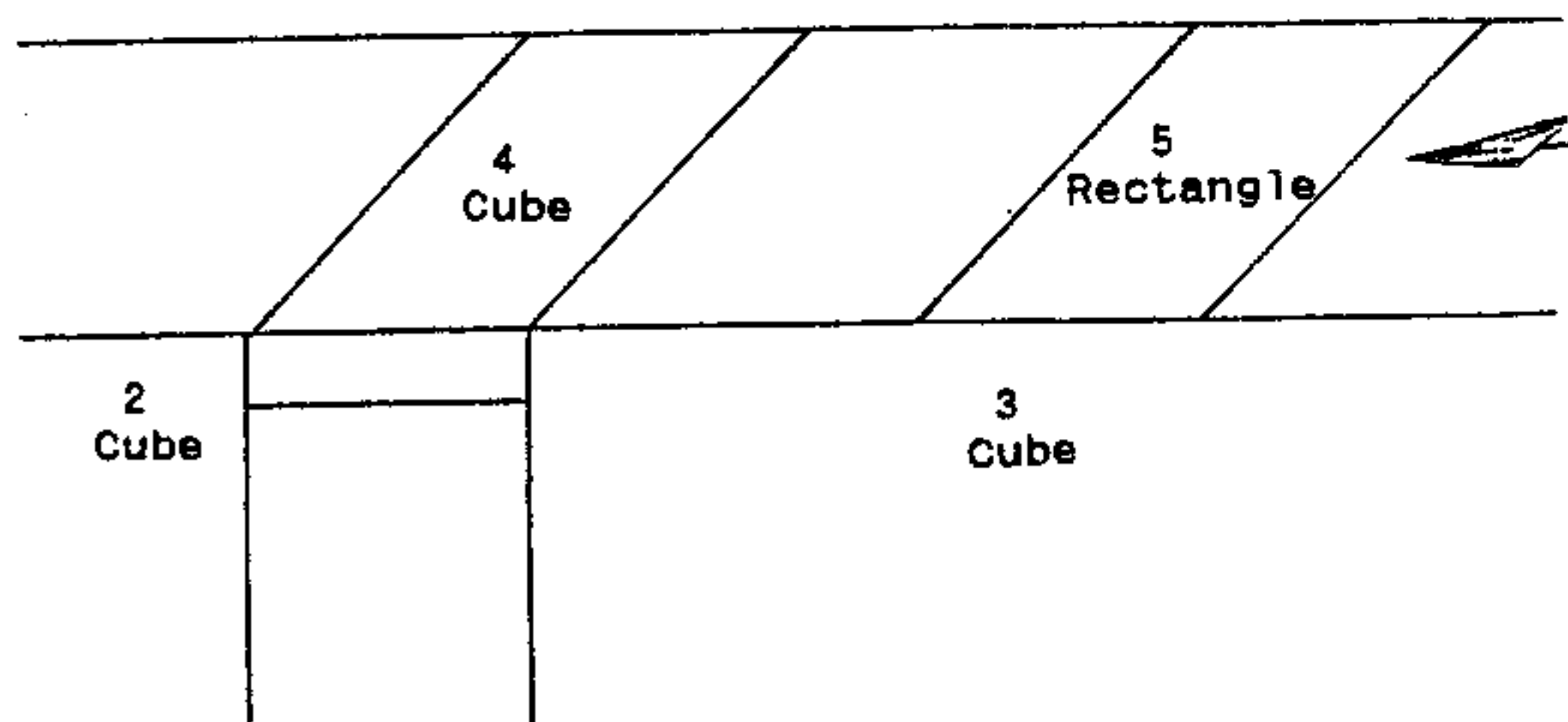
Reset the machine and insert your disk and (if you've done everything correctly) the disk should autoboot and display the startup screen (PIC1) closely followed by the instruction screen (PIC2) followed by your tune and your game. The tune will continue to play in the background while you play the game. There is no need to UNSHOW PIC2 because when your Kit game has loaded it will assume screen priority and clear the screen of anything else that it was displaying. I hope the above tips will be of use to other members and that my instructions are understandable! Oh by the way, you will need 1 meg of memory to run a soundtrack with your program.

### HARDER THAN IT LOOKS

By Richard Horn - PC

It is known that many puzzles involve getting something from somewhere, to be able to pass on to the next section. My idea below makes the entrance to the next bit look easy, but it ends up being just as hard as before.

|               |                |                |
|---------------|----------------|----------------|
|               | CONDITIONS     |                |
| FOR OBJECT 3: | FOR OBJECT 4:  | FOR OBJECT 5:  |
| IF COLLIDED?  | IF COLLIDED?   | IF COLLIDED?   |
| THEN VIS (4)  | THEN INVIS (4) | THEN INVIS (4) |
| ENDIF         | ENDIF          | ENDIF          |



As you walk in from the left, the trapdoor will disappear for a second giving you enough warning. If you carry on walking left the trapdoor will reappear but you will still fall through if you try to walk over it. All that is needed is for you to invent a suitable solution to the puzzle, say, pulling a lever in another area and entering suitable conditions to prevent the trapdoor from disappearing if the lever is pulled prior to entering the current area.

### GETTING YOU STARTED

By Jurgen Thuro - ATARI ST

PLANNING: Get used to your 3D Kit by working on it as much as possible, until using it becomes as easy as walking. Try out all the examples in the Newsletters as this helps enormously. Start a collection of ideas and write them down. Ask your friends for ideas too. When you think you have enough for one or more games then start to think about your first real project. When you have thought of a project then see which of your ideas are best suited to it and try to create an interesting story round your ideas.

Before you start to create your first object or the first border your game should be already created on paper. This means that every detail should be worked out. When I started I didn't think this too important and this was my first mistake. I had either too big or too small objects, wrong colours, wrong controls, wrong conditions and so on. It would have been easy to avoid all this if I had planned it.

There are some very good hints in the first newsletter about colours and borders, and Atari ST users should think about this especially due to the split screen. A full screen border may use a lot of colours that you will have to provide in every area so you are not free to make use of every colour you want to. You may find it easier to use a kind of scale in your planning, I use x=10 to have the correct proportions and assume it to be 1 inch. My first object was 15x3x15 instead of X=150, Y=30, X=150. When you plan your world, make sure that your player is able to reach every area you think he should. Is there enough fuel? Is the level too high? etc. Include animation sequences such as swinging view if you enter a planet's atmosphere or if you are hit by something. Try to use only 40 visible objects at a time to improve the performance. Make invisible parts of objects/primitives by colouring them invisible. Objects can be made invisible by a single trigger or sensor, if you are too far away to see them eg. on an upper floor of a house or behind it.

VARIABLES: Make an extensive use of variables 0 to 21. Some of you, who try their first project may think that 225 variables (255-30 used by the system) is a lot and that you will never need this many but believe me, it is very easy to use a lot of them. Up to now I have finished about 2 percent of my project and have already used 50 variables. Very often you will have variables used as flags such as:  
 V200: Gun on = 1, Gun off = 2.  
 V201: Lazer weak = 1, Medium = 2, Strong = 3.  
 V202: Rope available = 1, No = 0.

Using variables like this is very easy but very soon there will be no variables left for use. Here is the way I use my variables: Each variable in the range of V31 to V255 uses 32 bits to code all values. When I need variables as flags I use the bits as follows:



0 = False, 1 = True.

no hope                  no medium laser                  not weak laser

EG. V200:    31 30 29 28 27.....5 → 4    3   2   1   0 - (Number of bit)

               0   0   0   0   0.....1   0   1   0   0   1 - (Information)

                                 ↑                                  ↑                                  ←

                                 autopilot on                  strong laser                  gun on

This means that I can use "one" variable to code 32 different kinds of flags. Of course information like amount of money, amunition, fuel and so on should be coded as usual. I am just using variables used as simple flags or carrying little information (as a maximum 32 different values). But how to use the bits? Heres how I do it:

To set a bit:    bit no

ORV (2, V200) eg. ORV (2s, V200) = ORV (8, V200) = 0...01000  
bit no. 31...43210

As you can see you simply use an ORV command and the duel system to set a value of 1 to bit number 3 of variable V200. If you now want to set the gun to on and laser to strong and autopilot to on this would meant you set the bits with the numbers 0, 3 and 5 at the same time. Therefore you add each value of the duel system as follows:

$$2_0 + 2_3 + 2_5 = 1 + 8 + 32 = 41$$

and use the command ORV(41,V200) to get

```

31.....5 4 3 2 1 0 - bit no.
0.....1 0 1 0 0 1 - value

```

To read a bit you use the ANDV command. For example if you want to know if the autopilot is on (ie. check bit number 5) you would enter:

```

25 =32
IF ANDV (V200,32) THEN..... ENDIF

```

The effect is a bitwise logical combination:

effect is a bitwise logical combination:

|                             |      |                                |
|-----------------------------|------|--------------------------------|
| 31 30 29                    |      | 5 4 3 2 1 0                    |
| { 0 0 0 ..... 1 0 1 0 0 1 } | ANDV | { 31 30 29 ..... 5 4 3 2 1 0 } |
| V200=41                     |      | 32                             |

Here the result is 1 which means that bit number 5 is set and therefore the autopilot is on. If you use the command:

```
IF ANDV (V200,16) THEN... ENDIF
```

the result would be 0 because bit number 4 ( $2^4 = 16$ ) was not set (no rope carried) and the IF clause would react accordingly. Make sure that your variable is placed on the LEFT HAND SIDE within the brackets, otherwise the result 0 or 1 would be the new value of V200 which would mean that all your set flag values are lost.

TIPS: Write down a list of how you use the variables as you go along so you don't forget what they do. If many objects use the same condition, write this condition once in the General conditions with variables and no exact values so you make it flexible. That way you can use this condition from every object by using the EXECUTE (n) command.

command.

\*\*\*\*\*

Members are cordially invited to send in their routines for inclusion in these pages, whether it be a simple or complicated animation, a longer article or just a few hints or tips, they will be most welcome. If you have found a way to do something then why not share it with the other members so that everyone can benefit from your experiences.

## 8 BIT ROUTINES

### SIMULATING FREE FALL

By *James Neill* - AMSTRAD CPC6128

You can use variables 112 to 120 for automatic movement around an area and I've used these to simulate free fall after jumping out of an aeroplane. First of all start at the top of the area looking down as if you have stopped while falling to the ground, this position should be the entrance to the area (make sure you are in FLY2 mode). Now take note of all the viewpoint values (X, Y and Z). Now move down as if getting closer to the ground, then take note of the viewpoint values again, move downwards again and take notes again and keep doing this until you hit the ground. You should end up with 15-16 viewpoints. Now the hard bit... You must now set variables 112-120 with the first viewpoint, lets start with viewpoint X in the first step of the free fall, the high byte goes in variable 113 and the low byte goes in 112. If viewpoint X = 3209 then high byte =  $3209/256 = 12$ . Low byte = remainder = 137. Use the SETV command to load 112 with 137 and to load 113 with 12. In my case the first part of the free fall is:

X = 3209    Y = 4031    Z = 3249

So I set the variables in the following way:

```

SETV 137 112 137
SETV 012 113 +12X256 = 3209
SETV 191 114 191
SETV 015 115 +15X256 = 4031
SETV 117 116 117
SETV 012 117 +12X256 = 3249
REDRAW Shows the new

```

Shows the new position on screen.

Remember the REDRAW else it will seem like you are missing steps. Of course this is only one step towards the ground there is another 15 or so to go. To call the free fall routine I used the following LOCAL condition in Area 1 (the aeroplane).

```
IFHIT 009      - If the player hits the door
THEN
GOTO 001 002   - Goto entrance 1 in area 2 which positions the
                player looking down towards the ground.
CALL 001       - These procedures contain the variable settings
CALL 002       for animating the free fall. You can fit a
CALL 003       number of steps in each procedure.
```

Here is a routine to turn the cross (the one which looks like + not the moveable one) on and off using the "C" key. It should be put in as a General Condition. It uses variable 111 as a toggle. Note that this routine also solves the problem of nested IF's!

|       |     |     |
|-------|-----|-----|
| CMPV  | 067 | 121 |
| IFEQ  |     |     |
| THEN  |     |     |
| ELSE  |     |     |
| END   |     |     |
| ENDIF |     |     |
| CMPV  | 001 | 111 |
| IFEQ  |     |     |
| THEN  |     |     |
| CROSS | 000 |     |
| SETV  | 000 | 111 |
| ELSE  |     |     |
| CROSS | 001 |     |
| SETV  | 001 | 111 |
| ENDIF |     |     |
| END   |     |     |

## EXTRA OPTION FOR CPC6128 USERS

By James Neill

The following progette works on the Amstrad disc version and is really only useful for the CPC6128 users as we are the only ones who display a screen during testing. The program is in BASIC and replaces the file DISC. It adds an extra option, which allows the user to change the screen which is shown in TEST mode by replacing KIT1282.DAT with the new screen. The new screen should be a standard screen dump. The down arrow can be achieved by pressing CNTRL/J. This amendment will make designing screens easier as previously you had to compile the game with the screen to see what it looked like.

```

10 ' 3D CONSTRUCTION KIT LOADER inc. CHANGE BORDER
20 ' NOTE : BORDER IS ONLY SHOWN ON 128K VERSION
30 ' BY JAMES NEILL
40 '
50 MODE 1:MEMORY &3FFF
60 PRINT "↓↓ 3D CONSTRUCTION KIT"
70 PRINT "↓↓↓ 1 128K EDITOR"
80 PRINT "↓ 2 64K ENVIRONMENT EDITOR"
90 PRINT "↓ 3 64K CONDITION EDITOR"
100 PRINT "↓ 4 FREESCAPE COMPILER"
110 PRINT "↓ 5 DATA DISK FORMATTER"
120 PRINT "↓ 6 CHANGE TEST SCREEN BORDER"
130 INPUT "↓↓↓ ENTER SELECTION (1-6)";A$
140 IF A$="1" THEN RUN"KIT128
150 IF A$="2" THEN RUN"KITENV
160 IF A$="3" THEN RUN"KITCOND
170 IF A$="4" THEN RUN"COMPILER
180 IF A$="5" THEN RUN"FORMAT
190 IF A$="6" THEN GOSUB 220
200 RUN
210 ' CHANGE TEST SCREEN BORDER
220 MODE 2
230 INPUT "ENTER NEW BORDER FILEMANE (FILENAME.EXT): ",FILE$
240 PRINT "INSERT BORDER DISC AND PRESS A KEY ...":CALL &BB18
250 MEMORY &3FFF
260 LOAD FILE$+"",&4000
270 PRINT "INSERT 3D CONSTRUCTION KIT DISC AND PRESS A KEY ...":CALL
&BB18
280 !ERA,"KIT1282.DAT"
290 SAVE "KIT1282.DAT",B,&4000,&4000
300 PRINT "NEW BORDER IS INSTALLED PRESS A KEY FOR MENU...":CALL &BB18
310 RETURN
To save this program type SAVE"DISC." remember the full stop after
disc else it will be saved as DISC.BAS and when you try to run it
DISC. will run instead.

```

## INFORMATION SCREEN

By D.G. Marley - SPECTRUM +2

After reading Issue 2, I had an idea about the Information Screen that was mentioned. The variable 124 stores the area you are in, so place the following into a General Condition. We will use area 14 for the Information Screen:

```

CMPV 73 121 - if I is pressed
IFEQ
AND
CMPV (1) 124 - I in area 1
IFEQ
THEN
SETV (1) 100 - stores area that I was pressed in

```

GOTO 1 14

ENDIF

Repeat the above procedure, increasing the number in brackets until all the areas are covered. Now when I is pressed you will goto Area 14 and the area you were in is stored in variable 100. Now go to area 14 and enter the following into a Local Condition:

```

CMPV 73 121 - if I is pressed
IFEQ
AND
CMPV (1) 100 - checks what area you called the
information screen from

```

IFEQ

THEN

GOTO 1 (1)

ENDIF

Repeat the above procedure, increasing the number in brackets until all areas are covered, include messages at will. The only problem with this is you can only return to entrances you have placed in these areas. Hopefully someone else may have an idea on how to shorten it or make it possible to return to the position you pressed "I", or alter it to work on 16-bit machines.

The following procedure is useful for Title Screens, just to make them that more interesting, it moves the characters printed on the screen up and down:

```

TEXTCOL 2
PRINT 1 0 10
GOTO 1 1
ADDV 1 126
CMPV 2 126 NOTE: Create this procedure in area 2
IFEQ as a Local Condition, create message
SETV 0 126 and enter "Press ENTER to start".
ENDIF
CMPV 13 121
IFEQ
THEN
SETV 0 126
SETV 61 127

```

## TWO COLOUR INSTRUMENTS

By A.P. Smith - AMSTRAD CPC6128

Here is a routine to create a two-colour instrument. Set two instruments to lie alongside each other then use:

```

IFTIMER
THEN
SUBV 001 001
ENDIF
CMPV 254 001
IFGT
THEN
SETV 000 001
SUBV 001 002
ENDIF
CMPV 254 002
IFGT
THEN
SETV 000 002
ENDIF
END

```

\*\*\*\*\*



## PROBLEM PAGES

**Dear Mandy**

While reading the manual and studying the 3D Kit, I cannot see the difference between SAVE DATAFILE, SAVE OBJECT and SAVE POSITION as on the test screen in the game. Every time I created and edited an object I just cannot save it. Must I first format a disk of my own? Sometimes I also get a DOS.ERROR like: File Not Found. When I then turn on my computer and try to load the object it has vanished. Please can you tell me how to save an object in the right way?

**A.C.M. Tammer**, Holland - AMIGA

For your problem with saving and loading OBJECTS - first you should, as you suggest, have your own formatted disk ready to save your data to. As an example I suggest you create two pyramids one above the other and flip the lower one and join them together to form one shape (rather like a diamond shaped stone). Then create a group and select the two pyramids into the group. Now go to the file menu and select SAVE OBJECT. You will be presented with the file selector. At this point you should either insert your formatted disk or, if it is in a second drive, click on DF1: and click on the lower FILE (below PATH) and delete whatever is there and enter your own filename - say PYRAMIDS. Now click on OK and your object will be saved to your save disk. Remember to ensure that the disk you wish to save your object to is selected by clicking on either DF0: or DF1: - if it is just an empty initialised disk the file selector should show only two items - TRASHCAN and TRASHCAN.INF. After the save it will show PYRAMIDS, TRASHCAN and TRASHCAN.INF.....Mandy

**Dear Mandy**

Occasionally rooms "freak out" - no matter which direction I move in I keep seeing the same object overlaid on itself lots of times, it is difficult to explain but it has happened quite a few times. Most notably it usually occurs if I deleted the initial cuboid in a room.

**Matthew Clementson**, Birmingham - ATARI ST

What has obviously happened is that you have turned your HORIZON OFF. Never do that unless you are working in a completely enclosed area, ie. after having selected all the global objects (all four walls, roof and floor) otherwise strange effects will happen.....Mandy

**Dear Mandy**

When I create an area I find it impossible to change the scale properly. I also find it impossible to load the Kitgame and other items from file selectors because I keep getting a dialogue box telling me that it is not the correct format. When I select FLY 2 or CAMERAS 1-5 I cannot get out of the horizontal level. Please help.

**Paul Jarleton**, France - ATARI ST

Make sure that you don't change the scale of the area beyond about 12 for it to work properly. You must load the Kitgame from within the data FOLDER not 3DKit.File from the root directory. Fly 2 and the Camera modes force forward and backward movement to remain parallel to the horizon. This isn't a bug but is an intended attribute of these movement modes. To move out of the horizontal plane use up/down arrows and you should have no further trouble.....Mandy

**Dear Mandy**

I have just received my 3D Kit through the post. On trying it out I find that clicking on the arrows to move an object moves the object in the opposite direction to that indicated, up down or sideways. I know this is not a great problem but as it is new it would be nice if it worked correctly. Should I get the disk changed?

**M.C. Loveland**, Exeter - ATARI ST

I have checked the disc and there is no problem, Mr Loveland. I think the answer lies in a little confusion on the function of the arrows. The set of directional arrows which are always on screen are to control YOUR movement around the environment. If you create an object and select edit object a second set of icons with directional arrows will appear and it is these that will move your object. If you use the top set of arrows it will, indeed, appear as if the object is moving in a contra direction as it is your viewpoint that is moving. I can well understand the initial confusion (I had it myself) but with a little practice you will soon become totally familiar with all forms of movement.....Mandy

**Dear Mandy**

I am having problems when creating borders for 3D Construction Kit. I draw my borders on Deluxe Paint III, as NTSC, 16 colours, 320 x 200 pixels and Lo-res format, and save then under the name Border.IFF on a blank disk. Next I load up 3D Construction Kit and after selecting LOAD BORDER and the border I created, an error saying "The selected file is not of the correct format" appears. When I select Degas or Neo format, my border loads but looks very distorted.

**Jill King**, Huntingdon - AMIGA

It is much better if you save your borders as a BRUSH instead of a picture, Jill. If you save a screen in PAL format the program will save it at minimum size which is 320 x 256, even if you set the page size to 320 x 200. Saving as a brush prevents that happening.....Mandy

**Dear Mandy**

I am writing to solve David Turner's problem of using the demonstration game (Issue 2, page 16), as your reply wasn't exactly mind-blowing. To view the Kitgame you must have the condition editor loaded. This is on side two of tape 1, or if you have the Kit on disk then select condition editor from the boot menu. Now select LOAD and select file 9. The Kit demo is on side 2 of tape 2. If you have disk then the demo will load automatically. The demo will now have loaded and you can select TEST to see what the final game would be like, except no border will be displayed. To complete the demo as a complete game on cassette, load the compiler which is on tape 2 side 2. Now select file 9 to load and insert tape 2 side 1. Press play. The border is saved immediately after the demo, so leave tape 2 side 1 in the cassette player. When the compiler asks you if you want a border, press Y and type in KITSCR as the filename. Press play and the border will load. The compiler will now allow you to save the whole lot on a blank tape. If you are using disk then you will have a problem as there is a bug in the compiler, it will not allow you to load a border from disk. To solve this problem you have to transfer the file to tape. You will need a machine code monitor for this or a cartridge such as the Action Replay or Power Cartridge. Turn your C64 off and on again to clear the memory. Insert the 3D Kit disk and type LOAD"KITSCR",8,1 and when it has loaded, enter your monitor and type T 2000 4720 6000 T 6000 8720 3800 S",1,3800,5F20 Insert a blank cassette and press RECORD. When you enter the compiler load the demo from disk and the border from tape.

**Steven Flanagan**, Sheffield - COMMODORE 64

Thank you also for telling me about the "bug", Steven, which I will pass to the programmers for their attention.....Mandy

A PROBLEM SHARED IS OFTEN A PROBLEM SOLVED!

Don't suffer in silence, write in with your problems - you never know, someone else might just need the answer too but was too shy to ask!

## HINTS AND TIPS

TIPS by *J.C. Latty* - AMIGA

One irritating feature when using 3DKit on a hard drive is the fact that it was programmed to expect to be in df0: all the time. There is a way round it but not for the faint hearted and always on a backup copy. If you have a Hex file editor like NewZap or similar and scan the ASCII strings till you find the line that says df0: just change it to dh0: save it out and voila! no more irritating dos error messages.

One feature I find invaluable which wasn't mentioned in the manual is the ability to save a screen shot by pressing F2.

### COPYING AN AREA

By *Paul Ramsey* - AMIGA

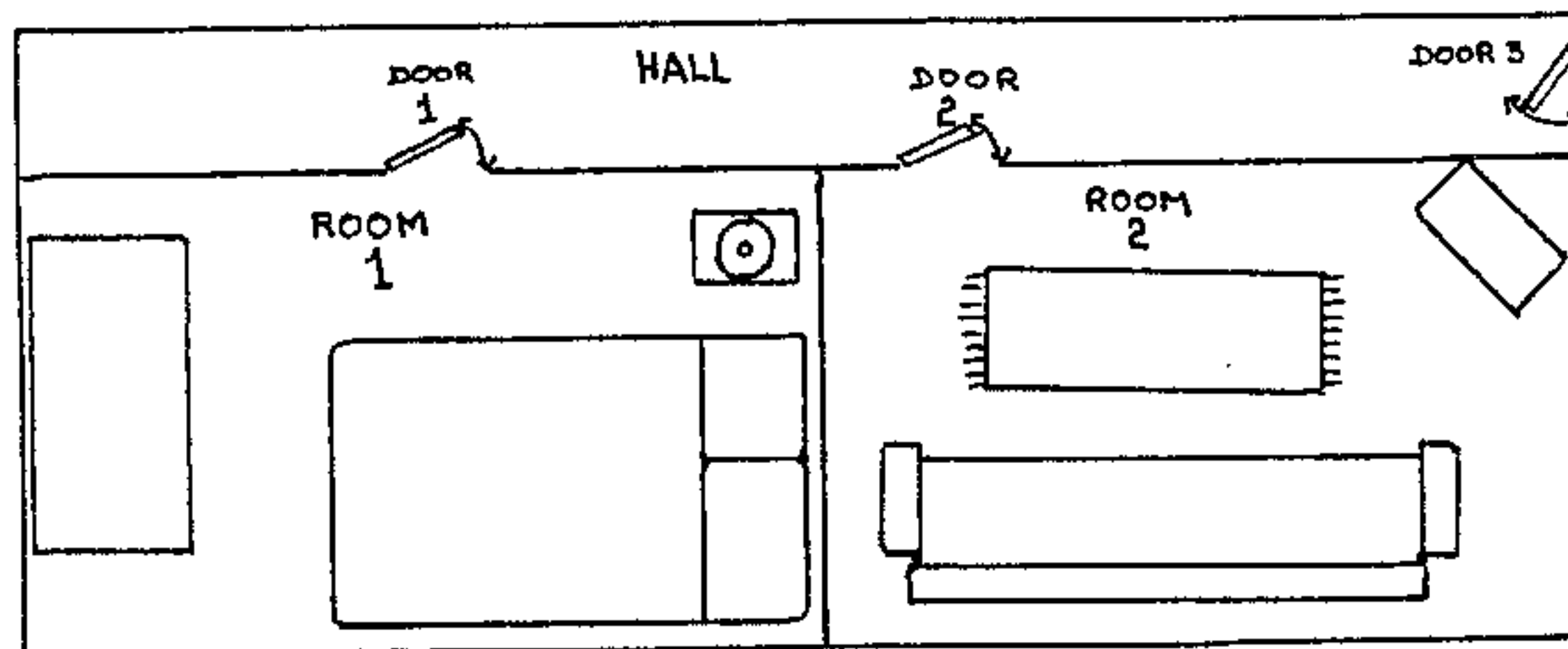
It is really quite simple to copy an area. What you should do is to make a GROUP which includes ALL the objects in your area and then save it as an OBJECT. Next you create another area and simply load in your OBJECT and the area is "copied".

### TO IMPROVE PERFORMANCE

By *Paul Gregory* - Programmer

If your area has many objects but some cannot be seen from certain positions you can use conditions to make them visible and invisible at certain times, ie.

#### OVERHEAD VIEW OF AREA



When door 2 is opened make all furniture in room 2 visible and all furniture in room 1 invisible. On exiting room 2 make all furniture in room 2 invisible and close the door and similarly for room 1 and door 1.

### HINT FOR DRAWING BORDERS

By *Mieke Van der Poel*

I believe I have a hint for everyone, including myself, who is not so hot on drawing borders. Try using an overhead transparent sheet with quick drying waterproof ink to trace borders from flight simulators, racing games etc, by sticking it over the screen and tracing it. When you have done this you can load in an art package and draw the same picture on the screen (you can see through the transparent sheet to do this). Then remove the sheet and colour in as you wish. It may not be exactly legal but it will help train you to draw your own borders.

ALL HINTS AND TIPS ARE WELCOME FOR THIS SECTION PLEASE